Samantha Gregoryk
CPTS 315
September 19th, 2020
Ananth Jillepalli

CptS 315: Introduction to Data Mining Homework 1 (HW1)

Q1. (25 points) Consider the following market-basket data, where each row is a basket and shows the list of items that are part of that basket.

1. *{A,B,C}*
2. *{A,C,D,E}*
3. *{A,B,F,G,H}*
4. *{A,B,X,Y,Z}*
5. *{A,C,D,P,Q,R,S}*
6. *{A,B,L,M,N}*

a) What is the absolute support of item set *{A,B}* ?

1,3,4,6 all contain {A, B} so the absolute support would be 4.

b) What is the relative support of item set *{A,B}* ?

1,3,4,6 all contain {A, B} out of the six options so the relative support would be $\frac{4}{6}$ or 0.66.

c) What is the confidence of association rule $A \Rightarrow B$?

1,3,4,6 all contain {A, B} and every basket contains {A} so the confidence would be $\frac{4}{6}$ or 0.66.

Q2. (15 points) Answer the below questions about storing frequent pairs using triangular matrix and tabular method.

a) Suppose we use a triangular matrix to count pairs and the number of items n = 20. If we store this triangular matrix as a *ragged* one-dimensional array Count, what is the index where count of pair (7,8) is stored?

(i, j) -> (7,8)

= (i – 1) (n - $\frac{i}{2}$) + (j – i)

(7 – 1) (20 - $\frac{7}{2}$) + (8 – 7) = (6)(16.5) + (1) = 100

Index where count of pair (7, 8) is stored is 100.

b) Suppose you are provided with the prior knowledge that only ten percent of the total pairs will have a non-zero count. In this case, which method among triangular matrix and tabular method should be preferred and why?

The tabular method would be preferred because it will intake less memory (12 bytes per pair) than the triangular method since there will be only ten percent of total pairs that will have a non-zero count, which is not greater than 1/3.

Q3. (35 points) This question is about the PCY algorithm for counting frequent pairs of items. Suppose we have six items numbered 1, 2, 3, 4, 5, 6. Consider the following twelve baskets.

1. {1,2,3}
2. {2,3,4}
3. {3,4,5}
4. {4,5,6}
5. {1,3,5}
6. {2,4,6}
7. {1,3,4}
8. {2,4,5}
9. {3,5,6}
10. {1,2,4}
11. {2,3,5}
12. {3,4,6}

Suppose the support threshold is 4. On the first pass of the PCY algorithm, we use a hash table with 11 buckets, and the set *{i,j}* is hashed to *i* × *j* mod 11.

a) By any method, compute the support for each item and each pair of items.

I found the support by counting how many times each item occurred in all the baskets.

s({1}) = 4	s({2}) = 6	s({3}) = 8	s({4}) = 8	s({5}) = 6	s({6}) = 4
s({1,2}) = 2	s({1,3}) = 3	s({1,4}) = 2	s({1,5}) = 1	s({1,6}) = 0	
s({2,3}) = 3	s({2,4}) = 4	s({2,5}) = 2	s({2,6}) = 1		
s({3,4}) = 4	s({3,5}) = 3	s({3,6}) = 2			
s({4,5}) = 3	s({4,6}) = 3				
s({5,6}) = 2					

b) Which pairs hash to which buckets?

Pair {i, j} hashes to bucket i x j mod 11:

{1,2} = 2	{1,3} = 3	{2,3} = 6
{2,3} = 6	{2,4} = 8	{3,4} = 1
{3,4} = 1	{4,5} = 9	{3,5} = 4
{4,5} = 9	{5,6} = 8	{4,6} = 2
{1,3} = 3	{3,5} = 4	{1,5} = 5
{2,4} = 8	{4,6} = 2	{2,6} = 1
{1,3} = 3	{3,4} = 1	{1,4} = 4
{2,4} = 8	{4,5} = 9	{2,5} = 10
{3,5} = 4	{5,6} = 8	{3,6} = 7
{1,2} = 2	{2,4} = 8	{1,4} = 4
{2,3} = 6	{3,5} = 4	{2,5} = 10
{3,4} = 1	{4,6} = 2	{3,6} = 7

Bucket	Pair
1	{2,6}, {3,4}
2	{1,2}, {4,6}
3	{1,3}
4	{1,4}, {3,5}
5	{1,5}
6	{1,6}, {2,3}
7	{3,6}
8	{2,4}, {5,6}
9	{4,5}
10	{2,5}

c) Which buckets are frequent?

From the table above:

Bucket 1: count = 5

Bucket 2: count = 5

Bucket 3: count = 3

Bucket 4: count = 6

Bucket 5: count = 1

Bucket 6: count = 3

Bucket 7: count = 2

Bucket 8: count = 6

Bucket 9: count = 3

Bucket 10: count = 2

The count needs to be greater than 4 in the first pass to be frequent. Since buckets 1, 2, 4, and 8 all have a support larger than 4, they are frequent buckets.

d) Which pairs are counted on the second pass of the PCY algorithm?

There are two conditions that need to be satisfied in order to move on to the second pass of the PCY algorithm which are (1) both *i* and *j* are frequent items and (2) the pair {*I,j*} hashed to a frequent bucket (whose bit in a vector is 1).

- Condition 1 satisfied?
  - Yes, all items are frequent in the buckets.
- Condition 2 satisfied?
  - Yes, but only pairs {1,2}, {1,4}, {2,4}, {2,6}, {3,4}, {3,5}, {4,6}, {5,6} can move on to the second pass of the PCY algorithm since they have a count larger than the support threshold of 4.

Q4. (25 points) Please read the following paper and write a brief summary of the main points in at most ONE page. You can skip the theoretical parts.

Saul Schleimer, Daniel Shawcross Wilkerson, Alexander Aiken: Winnowing: Local Algorithms for Document Fingerprinting. SIGMOD Conference 2003: 76-85

https://theory.stanford.edu/~aiken/publications/papers/sigmod03.pdf

With the large amounts of information provided from the internet, anyone can search almost anything they want and find an answer. In the last decade, plagiarism has become an issue that just increases as more items become available on the internet. Document fingerprinting is a way to keep track of misused work and be able to report these findings to the appropriate source. Copy-detection algorithms like local and winnowing should be able to cover whitespace insensitivity, noise suppression, and position independence.

The winnowing algorithm, analyzes performance on random data, with a given window size w, the density is asymptotically 2/(w+1), it proves lower bound of 1.5/(w+1) on the density of local algorithms**,** and is 33% optimal. The basic implementation for winnowing follows the guidelines of in each window select the minimum hash value. If there is more than one hash with the minimum value, select the rightmost occurrence. Now save all selected hashes as the fingerprints of the document. This algorithm is beneficial because it gives us some flexibility to treat the two fingerprinting times (database-build time and query time) differently.

Local fingerprinting algorithms can be an alternative for the winnowing algorithm based on the dependence of external information about the position of the window in the file (or its relationships) to other windows. S is a selection function taking a w-tuple of hashes and returning an integer between zero and w−1, inclusive. A fingerprinting algorithm is local with selection function S, if, for every window h_i,…h_{i+w-1}, the hash at position *i + S*(h_i,…h_{i+w-1}) is selected as a fingerprint.

Experiments done through the World Wide Web that generated random text are simply to check and see if the hash functions used can be trusted. These trials are important to make sure all information being collected is accurate and can be trusted. Another experiment was to calculate the hases of HTML documents and measure various statistics by calculating the density with the expected density for both winnowing and selecting fingerprints. Services like MOSS use all kinds of data from internet to track plagiarism.

Overall, plagiarism is the comparison of similarity and can be referenced from source all over the World Wide Web. For companies like MOSS to be able to track their detections, the system needs to work in practice and be able to be efficient all the time. Lastly, presentation needs to be readable to the user and show the matches that were detected in the text.